

The invention in which an exclusive right is claimed is defined by the following:

1. A method for using activity based notation to define interactions between a system and a user, comprising the steps of:

(a) separating the interaction between a user and a system into a plurality of types of interactions, including:

(i) inputter based interactions that involve data provided by the user to the system;

(ii) outputter based interactions that involve data provided by the system to the user;

(iii) invoker based interactions that involve an action taken by the user to change a state of the system, and which do not involve an exchange of data apparent to the user; and

(iv) selector based interactions that involve at least one item of data being provided to the user by the system, and a subsequent selection of at least one such item of data by the user;

(b) generating a statement for each interaction between a user and a system, each statement containing elements providing information required to completely describe the type of interaction and a nature of any information exchanged between the user and the system as a consequence of the interaction, said elements including at least:

(i) a symbol indicating the type of interaction;

(ii) a textual description of the interaction;

(iii) a definition of a type of data exchanged between the user and the system; and

(iv) a definition of a number of items of data exchanged during the interaction.

2. The method of Claim 1, wherein the step of generating the statement comprises the step of including in the statement any filters defining restrictions upon the data exchanged between the user and the system.

3. The method of Claim 1, wherein the step of generating the statement comprises the step of including in the statement any conditions that must be met by the data exchanged between the user and the system, in accordance with predefined system rules.

4. The method of Claim 1, wherein the definition of the number of items of data exchanged during the interaction indicates which items of data are optionally exchanged and which items of data that are required to be exchanged.

5. The method of Claim 1, wherein the step of generating the statement comprises the step of generating the statement according to rules that define relative positions of each element within the statement.

6. The method of Claim 1, further comprising the step of including the statements in a flow diagram.

7. The method of Claim 6, wherein said flow diagram comprises at least one of an activity diagram and a flowchart.

8. The method of Claim 6, further comprising the step of automatically generating a graphical user interface (GUI) form for guiding the user through each interaction with the system, said GUI form including at least one group for each statement.

9. The method of Claim 8, wherein the step of automatically generating the GUI form comprises the steps of:

(a) mapping a GUI form to the flow diagram, such that each different statement in the flow diagram is separately mapped to a different group in the GUI form; and

(b) labeling each group in the GUI form based on a corresponding statement.

10. The method of Claim 1, further comprising one of the steps of generating and modifying a graphical user interface (GUI) form, used for guiding the user through each interaction with the system, as defined by the generated statements.

11. The method of Claim 10, further comprising the step of automatically modifying a flow diagram describing each interaction between the user and the system, as the GUI form is modified.

12. The method of Claim 10, further comprising the step of automatically generating a flow diagram describing each interaction between the user and the system, as the GUI form is generated.

13. The method of Claim 12, wherein the step of automatically generating a flow diagram comprises the steps of:

- (a) mapping the GUI form to the flow diagram;
- (b) ensuring that each interaction shown in the GUI form is present in the flow diagram; and
- (c) for each interaction in the GUI form, ensuring that information from statements corresponding to each activity in the GUI form are included in a corresponding interaction in the flow diagram.

14. The method of Claim 13, further comprising the step of prompting a user to identify any statement information not automatically recognized.

15. The method of Claim 6, further comprising the step of automatically generating test scripts based on the flow diagram.

16. The method of Claim 15, wherein the step of automatically generating test scripts based on the flow diagram comprises the steps of:

- (a) parsing the flow diagram such that each statement corresponding to a different interaction between the user and the system is identified;
- (b) providing a graphical user interface (GUI) form used for guiding the user through each interaction with the system and parsing diagram mapping information that maps the flow diagram to the GUI form;
- (c) parsing the GUI form to identify individual GUI components;
- (d) mapping each GUI component to a statement in the flow diagram;
- (e) for each GUI component, parsing the statement mapped to that GUI component; and
- (f) generating a test script for that GUI component.

17. The method of Claim 16, further comprising the step of executing each test script to determine if the GUI form is displayed properly.

18. The method of Claim 17, wherein the step of executing each test script to determine if the GUI form is displayed properly comprises the steps of:

- (a) parsing the flow diagram to identify each statement corresponding to a different interaction between the user and the system;
- (b) mapping the GUI form to the flow diagram;
- (c) retrieving and parsing each test script corresponding to the flow diagram;
- (d) displaying the GUI form;
- (e) selecting a GUI component from the GUI form;
- (f) identifying a portion of the flow diagram corresponding to the GUI component selected;
- (g) parsing each path in the portion of the flow diagram corresponding to the GUI component selected by the user, such that paths corresponding to interaction types are identified and parsed to identify corresponding statements and test scripts;
- (h) identifying the interaction type and performing the indicated interaction;
- (i) executing the test script to determine if an error code results; and
- (j) logging any resulting error code.

19. The method of Claim 18, wherein if a plurality of parameters can apply to affect the test script, the test script is executed for each permutation and combination of parameters that can apply to the test script.

20. The method of Claim 18, further comprising the step of determining if the GUI form displayed includes any GUI components for which a test script has not been executed, and if so, selecting that GUI component and implementing steps (f)-(j) of Claim 17.

21. The method of Claim 18, further comprising the step of determining if an additional GUI form is being displayed, and if so, selecting a GUI component from the additional GUI form, and implementing steps (f)-(j) of Claim 17 for the GUI component selected from the additional GUI form.

22. The method of Claim 18, wherein if the interaction type identified in step (h) of Claim 17 is an inputter based interaction, further comprising the step of inputting random data before executing the script.

23. The method of Claim 18, wherein if the interaction type identified in step (h) of Claim 17 is an outputter based interaction, further comprising the steps of parsing the output, and applying any filters and conditions before executing the script.

24. The method of Claim 18, wherein if the interaction type identified in step (h) of Claim 17 is an invoker based interaction, further comprising the steps of invoking the interaction, and applying any filters and conditions before executing the script.

25. The method of Claim 18, wherein if the interaction type identified in step (h) of Claim 17 is a selector based interaction, further comprising the steps of generating all possible selection sets, and applying any filters and conditions to each selection set before executing the script for each selection set.

26. The method of Claim 6, further comprising the step of performing a simulation of the flow diagram to enable the user to determine if a GUI form for guiding the user through each interaction with the system is correct.

27. The method of Claim 26, wherein the step of performing a simulation of the flow diagram to determine if the GUI form is correct comprises the steps of:

- (a) parsing the flow diagram such that each statement corresponding to a different interaction between the user and the system is identified;
- (b) mapping the GUI form to the flow diagram;
- (c) displaying the GUI form;
- (d) selecting a GUI component from the GUI form;
- (e) identifying a portion of the flow diagram corresponding to the GUI component selected;

(f) parsing each path in the portion of the flow diagram corresponding to the GUI component selected by the user, such that paths corresponding to interaction types are identified and parsed to identify corresponding statements;

(g) identifying the interaction type and performing the indicated action to produce an updated GUI form; and

(h) displaying the updated GUI form to enable the user to determine if it is correct.

28. The method of Claim 27, wherein if the interaction type identified in step (g) of Claim 25 is an inputter based interaction, further comprising the steps of inputting random data before displaying the updated GUI form.

29. The method of Claim 27, wherein if the interaction type identified in step (g) of Claim 25 is an outputter based interaction, further comprising the steps of parsing the output, and applying any filters and conditions before displaying the updated GUI form.

30. The method of Claim 27, wherein if the interaction type identified in step (g) of Claim 25 is an invoker based interaction, further comprising the steps of invoking the action, and applying any filters and conditions before displaying the updated GUI form.

31. The method of Claim 27, wherein if the interaction type identified in step (g) of Claim 23 is a selector based interaction, further comprising the steps of generating all possible selection sets, and applying any filters and conditions to each selection set before displaying the updated GUI form for that selection set.

32. The method of Claim 27, wherein if a plurality of parameters can apply to affect the interaction type, further comprising the step of implementing each permutation and combination of parameters that can apply to the interaction type, to produce an updated GUI form for each such different permutation and combination.

33. The method of Claim 27, further comprising the step of determining if the GUI form displayed includes any GUI components for which an updated GUI form has not been produced, and if so, selecting that GUI component and implementing steps (e)-(h) of Claim 23.

34. The method of Claim 27, further comprising the step of determining if an additional GUI form is being displayed, and if so, selecting a GUI component from the additional GUI form, and implementing steps (e)-(h) of Claim 26 for the GUI component selected from the additional GUI form.

35. The method of Claim 6, further comprising the step of producing graphical user interface (GUI) hardware components from a computer aided design (CAD) drawing, the GUI hardware components being configured to guide the user through each interaction with the system.

36. The method of Claim 35, wherein the step of producing GUI hardware components comprises the steps of:

- (a) mapping a CAD drawing to the flow diagram, such that each different statement in the flow diagram is separately mapped to a different group in the CAD drawing;
- (b) labeling each group in the CAD drawing based on a corresponding statement; and
- (c) using the CAD drawing to control equipment to produce the hardware components.

37. The method of Claim 1, further comprising the steps of:

- (a) providing a computer aided design (CAD) drawing of a graphical user interface (GUI) hardware component configured to guide the user through each interaction with the system; and
- (b) automatically generating a flow diagram describing each interaction between the user and the system based on the GUI hardware component.

38. The method of Claim 37, wherein the step of automatically generating a flow diagram comprises the steps of:

- (a) mapping the CAD drawing to a flow diagram;
- (b) ensuring that each interaction shown in the CAD drawing is present in the flow diagram; and
- (c) for each interaction in the CAD drawing, ensuring that information from statements corresponding to each interaction in the CAD drawing are included in the corresponding interaction in the flow diagram.

39. The method of Claim 6, further comprising the step of quantifying a number of action states in the flow diagram.

40. The method of Claim 39, wherein the flow diagram includes a plurality of blocks, at least some of which define at least one action state, and wherein the step of quantifying a number of action states in the flow diagram comprises the steps of:

(a) parsing the flow diagram to identify each block in the flow diagram that defines at least one action state;

(b) for each block defining an action state, determining if that block includes a statement corresponding to an interaction between the user and the system, and if so, determining a number of statements in that block;

(c) determining a number of blocks that define at least one action state and do not include such a statement; and

(d) combining the number of blocks that define at least one action state and do not include such a statement with the number of statements in each block defining an action state that includes such a statement, to quantify the number of action states in the flow diagram.

41. The method of Claim 6, further comprising the step of determining a scope of the flow diagram.

42. The method of Claim 41, wherein the flow diagram corresponds to at least one of a software based system and a hardware based system.

43. The method of Claim 41, wherein the step of determining a scope of the diagram comprises the steps of quantifying a number of action states in the flow diagram, and evaluating a level of effort associated with the flow diagram.

44. The method of Claim 43, wherein the flow diagram comprises an activity diagram including a plurality of swimlanes, and wherein the step of evaluating a level of effort associated with the flow diagram comprises the steps of:

(a) parsing the flow diagram to determine a number of paths contained in the flow diagram;

(b) counting the plurality of swimlanes to determine a number of swimlanes in the flow diagram;



(c) identifying a number of crossings between the plurality of swimlanes; and

(d) using the number of paths in the flow diagram, the number of swimlanes in the flow diagram, and the number of crossings between swimlanes to evaluate a level of effort associated with the flow diagram.

45. The method of Claim 43, wherein the flow diagram comprises a flowchart, and wherein the step of evaluating a level of effort associated with the flow diagram comprises the steps of:

(a) parsing the flow diagram to determine a number of paths contained in the flow diagram; and

(b) using the number of paths in the flow diagram to evaluate a level of effort associated with the flow diagram.

46. A system configured to use activity based notation to enhance a design and evaluation of a system configured to interact with a user, where the activity based notation defines interactions between a system and a user, comprising:

(a) a computing device including:

(i) an input device that receives input from a user;

(ii) a memory in which machine instructions and data are stored;

(iii) a display; and

(iv) a processor coupled to the input device, the memory and the display, said processor executing the machine instructions to carry out a plurality of operations, including:

(1) enabling a user to separate each interaction between a user and a system into one of the following four types of interactions;

(A) inputter based interactions that involve data provided by the user to the system;

(B) outputter based interactions that involve data provided by the system to the user;

(C) invoker based interactions that involve an action taken by the user to change a state of the system, and which do not involve an exchange of data apparent to the user; and

(D) selector based interactions that involve at least one item of data being provided to the user by the system, and a subsequent selection of at least one such item of data by the user; and

(2) generating a statement for each interaction between a user and a system, each statement containing elements providing information required to completely describe the type of interaction and a nature of any information exchanged between the user and the system as a consequence of the interaction, said elements including at least;

(A) a symbol indicating the type of interaction;

(B) a textual description of the interaction;

(C) a definition of the type of data exchanged between the user and the system; and

(D) a definition of a number of items of data exchanged during the interaction.

47. The system of Claim 46, wherein the machine instructions further cause the processor to include in each statement any filters defining restrictions upon the data exchanged between the user and the system.

48. The system of Claim 46, wherein the machine instructions further cause the processor to include in each statement any conditions that must be met by the data exchanged between the user and the system, in accordance with predefined system rules.

49. The system of Claim 46, wherein the definition of the number of items of data exchanged during the interaction indicates which items of data are optionally exchanged and which items of data that are required to be exchanged.

50. The system of Claim 46, wherein the machine instructions further cause the processor to generate each statement according to rules that define relative positions of each element within the statement.

51. The system of Claim 46, wherein the machine instructions further cause the processor to include the statements in a flow diagram.

52. The system of Claim 51, wherein the machine instructions further cause the processor to use the statements in the flow diagram to automatically generate a graphical user interface (GUI) form for guiding the user through each interaction with the system, said GUI form including at least one group for each statement.

53. The system of Claim 52, wherein the machine instructions further cause the processor:

- (a) enable a user to make changes to the GUI form; and
- (b) in response to such changes in the GUI form, automatically update the flow diagram to reflect such changes.

54. The system of Claim 51, wherein the machine instructions further cause the processor to use the statements in the flow diagram to automatically generate test scripts.

55. The system of Claim 54, wherein the machine instructions further cause the processor to execute each test script to identify any error codes that may be produced when the test script is executed.

56. The system of Claim 51, wherein the machine instructions further cause the processor to use the statements in the flow diagram to automatically simulate an application defined by the flow diagram and to display a GUI form based on the flow diagram, to enable a user to evaluate the GUI form.

57. The system of Claim 51, wherein the machine instructions further cause the processor to use the statements in the flow diagram to automatically generate a computer aided design drawing that can be used to produce hardware components for a GUI configured to guide the user through each interaction with the system.

58. The system of Claim 51, wherein the machine instructions further cause the processor to use the statements in the flow diagram to facilitate a quantification of a number of action states in the flow diagram.

59. The system of Claim 58, wherein the flow diagram includes a plurality of blocks, at least some of which define at least one action state, and wherein the wherein the machine instructions further cause the processor to:

(a) parse the flow diagram to identify each block in the flow diagram that defines at least one action state;

(b) for each block defining an action state, determine if that block includes a statement corresponding to an interaction between the user and the system, and if so, determining a number of statements in that block;

(c) determine a number of blocks that define at least one action state and do not include such a statement; and

(d) combine the number of blocks that define at least one action state and do not include such a statement with the number of statements in each block defining an action state that includes such a statement, to quantity the number of action states in the flow diagram.